TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky a mezioborových inženýrských studií

Studijní program: N2612 – Elektrotechnika a informatika
Studijní obor: 1802T007 – Informační technologie

Nástroje pro syndikaci obsahu

Tools for a content syndication

Diplomová práce

Autor: **Bc. Jan Holub**
Vedoucí práce: Doc. RNDr. Pavel Satrapa, Ph. D.

V Liberci 21. 5. 2010

Originál zadání práce

Prohlášení

Byl(a) jsem seznámen(a) s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 o právu autorském, zejména § 60 (školní dílo).

Beru na vědomí, že TUL má právo na uzavření licenční smlouvy o užití mé diplomové práce a prohlašuji, že **s o u h l a s í m** s případným užitím mé diplomové práce (prodej, zapůjčení apod.).

Jsem si vědom(a) toho, že užít své diplomové práce či poskytnout licenci k jejímu využití mohu jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

Diplomovou práci jsem vypracoval(a) samostatně s použitím uvedené literatury a na základě konzultací s vedoucím diplomové práce a konzultantem.

Datum

Podpis

Poděkování

Tímto bych chtěl poděkovat a vyslovit uznání všem, kteří přispěli ke vzniku této práce. Především vedoucímu mé diplomové práce panu Doc. RNDr. Pavlu Satrapovi, Ph. D. za množství praktických rad a věcných připomínek.

Dále bych chtěl poděkovat rodičům za podporu a poskytnuté zázemí v celém průběhu mého studia.

Abstrakt

Diplomová práce se věnuje metodám pro syndikaci obsahu na webu, pro kterou se používají dva základní formáty RSS a Atom. Oba mají celkem dlouhou historii, která je rozvedena v úvodu práce. V další části se zabývám detailním charakteristikou obou formátu, jejich strukturou a popisem všech elementů a atributů. Konkrétně se věnuji formátům RSS ve verzi 1.0 a 2.0 a formátu Atom ve verzi 1.0. Četnost použití jmenovaných syndikační formátů na webu je různá. Součástí práce jsou statistiky jejich rozšířenosti v celosvětovém i domácím měřítku. Ty jsou podloženy jak veřejně dostupnými statistikami, tak vlastnoručně získanými daty. Výstupem práce je multiplatformní aplikace napsaná v jazyce Java pro správu syndikačních formátů. Jedná se o jednoduchý editor zdrojů podporující formáty RSS a Atom. Aplikace umožňuje otevírat zdroje jak z lokálního disku, tak z URL. Dále pak přidávat, odebírat elementy a upravovat zdroj v integrovaném textovém editoru. Vedle těchto základních funkcí podporuje vzájemnou konverzi formátů a export zdroje do (X)HTML.

Abstract

The thesis deals with methods of content syndication. The two main families of web syndication formats are RSS and Atom. Both of them have a long history which is described on the beginning of this work. In the next part I describe the structure and all available elements and attributes of the both syndication formats with a view to the RSS format in version 1.0 and 2.0 and the Atom format in version 1.0. Usage frequency of these syndication formats on web sites is various. This paper includes a statistical data of the formats distribution at worldwide and national basis. The data are supported by public and self-obtained statistics. The output of this thesis is a multiplatform application written in Java for managing syndication formats. It is a simple editor of RSS and Atom feeds which can open these feeds from a local hard drive or from a URL. The application allows the user to add and remove elements and edit the feed in the integrated text editor. It also supports a conversion between syndication formats and export to (X)HTML.

Obsah

Seznam zkratek

RSS	Really Simple Syndication (původně Rich Site Summary)
URL	Uniform Resource Locator
URI	Uniform Resource Identifier
HTML	HyperText Markup Language
XHTML	Extensible HyperText Markup Language
XML	Extensible Markup Language
MCF	Meta Content Framework
KRL	Knowledge Representation Language
KIF	Knowledge Interchange Format
RDF	Resource Description Framework
CDF	Channel Definition Format
W3C	World Wide Web Consortium
API	Application Programming Interface
IETF	Internet Engineering Task Force
RFC	Request For Comments
CGI	Common Gateway Interface
MIME	Multipurpose Internet Mail Extensions
ISBN	International Standard Book Number
APP	Atom Publishing Protocol
HTTP	Hypertext Transfer Protocol
XML-RPC	Extensible Markup Language – Remote Procedure Call
BSD	Berkeley Software Distribution
ROME	RSS And Atom Utilities

1. Úvod

Internet a informační technologie obecně jsou bezpochyby jedny z nejrychleji se rozvíjejících odvětví. S tímto technologickým rozmachem souvisí obrovské množství informací, které je dnes možné na webu nalézt. Jejich objem se navíc zvyšuje každým dnem. Běžný uživatel internetu se tak může v této záplavě dat velmi snadno ztratit. Existuje však několik prostředků, které pomáhají čtenáři získat pouze informace, které ho zajímají, a jedním z nich je syndikace obsahu – webová služba využívající tzv. RSS a Atom kanály.

Syndikace obsahu je již delší dobu standardem ve světě velkých zpravodajských a informačních serverů. V dnešní době se však stala také součástí malých webů a blogů, které může provozovat prakticky kdokoliv bez větších znalostí webových technologii. Tento rozmach umožnily blogovací a redakční systémy nabízejí velké množství funkcí, kde mezi ty základní patří také automatické generování syndikačních kanálů.

Jako příklad využití těchto kanálů si představme uživatele internetu, který chce být neustále informován například o vývoji ekonomické krize. Existují desítky serverů, které poskytují ekonomické zpravodajství. Takový člověk by tedy musel pravidelně navštěvovat dané weby a ručně kontrolovat nové články, zda neobsahují informace, které ho zajímají. Takový postup je značně časově náročný. Lze ho však zefektivnit využitím syndikačních kanálů. Čtenář si pouze přidá odkazy na zdroje syndikovaného obsahu, které nalezne na příslušném webu, do speciálního programu nazývaného čtečka. Ta už se pak automaticky postará o pravidelnou kontrolu webu a nové články oznámí uživateli.

Každý zdroj syndikovaného obsahu musí být v předepsaném formátu. Dnes se můžeme setkat prakticky se dvěma základními formáty, které jsou založeny na XML. Jsou jimi výrazně rozšířenější RSS a méně oblíbený Atom. Čtenář v této práci nalezne jejich historii, popis a vzájemné porovnání. Jelikož oba formáty slouží ke stejnému účelu, nabízí se možnost jejich vzájemné konverze. Pro tyto účely vznikla multiplatformní aplikace, kterou se zabývám v praktické části práce. Ta mimo jiné umožňuje zmíněnou konverzi formátu a export zdroje do (X)HTML. Existují sice podobné aplikace umožňující správu syndikačních kanálu, ty jsou ovšem většinou placené a určené pouze pro systém Windows.

2. Historie syndikačních formátů

V této kapitole se budu věnovat vzniku a historii formátu pro syndikaci obsahu na webu. Nejdříve přiblížím celkem dlouhou a trnitou cestu RSS až k finální verzi 2.0 a poté se také zmíním o vzniku konkurenčního standardu Atom.

2.1. Historie formátu RSS

2.1.1. Předchůdci RSS

Počátky zrodu RSS spadají do roku 1995, kdy vznikl formát nazvaný Meta Content Framework (MCF), který vycházel ze systému CycL, KRL a KIF. Cílem MFC bylo popisovat objekty, jejich vlastnosti a také vztahy mezi jednotlivými objekty. Ke konci roku 1996 již existovalo několik stovek webů, které tento formát podporovaly. Přepsáním MCF do XML formátu vznikl nový formát nazvaný Resource Description Framework (RDF), který lze chápat jako obecný popisný jazyk pro reprezentaci informací na webu. RDF také tvoří základ pro dnes velmi aktuální a diskutovaný koncept tzv. sémantického webu[1].

V roce 1997 vytvořila firma Microsoft nový systém nazvaný Channel Definition Format (CDF), který byl přijat jako standard W3C. CDF bylo také založeno na XML a popisovalo jak obsah webu, tak jeho rozvržení a příslušná metadata. Formát CDF byl velmi zajímavou novinkou a zaujal hlavně nově se rodící komunitu web bloggingu a to navzdory tomu, že byl tento standard ve skutečnosti určen pro velké vydavatele. Mnoho elementů bylo zbytečně „silných“, a proto se někteří bloggeři pustili do vytváření jednodušší specifikace.

2.1.2. Vznik RSS

2.1.2.1. RSS 0.90 a 0.91

Roku 1998 představila firma Netscape svůj portál My Netscape, který měl zobrazovat aktuální zprávy a informace od různých vydavatelů. Systém byl založen na vlastním formátu Open-SPF. Vznikl tak první web (tzv. agregátor), který uživatelům poskytoval obsah oblíbených stránek na jednom místě. V roce 1999 byl

[1] Jedná se o web, kde jsou informace strukturovány a uloženy podle standardizovaných pravidel, což usnadňuje jejich vyhledání a zpracování.

formát Open-SPF přejmenován a vydán jako specifikace RDF-SPF 0.9. Nakonec došlo k další změně názvu a nový standard RSS 0.9 byl na světě. V témže roce byla vydána také první desktopová RSS čtečka nazvaná Carmen's Headline Viewer.

První koncept formátu RSS byl plně založen na datovém modelu RDF, a byl proto příliš komplikovaný pro koncové uživatele. Z tohoto důvodu neobsahoval nově vzniklý standard RSS 0.91 (rok 1999) žádné vlastnosti formátu RDF.

2.1.2.2. RSS 0.92 a 1.0

Za dalším vývojem standardu RSS již nestála firma Netscape, ale vývojářská komunita, která se ovšem rozdělila na dva tábory. První skupina chtěla do standardu vnést určitou formu rozšiřitelnosti. Ke konci roku 2000 tak vznikla nové verze RSS 1.0. Hlavními rysy bylo použití systému modulů, jmenných prostorů XML a návrat k datovému modelu RDF. Naopak druhá skupina měla obavy, že by takové zesložitění standardu nebylo běžnými uživateli vůbec přijato. Místo toho chtěla zachovat RSS jednoduché. A proto ve stejné době jako RSS 1.0 vznikla druhá verze RSS 0.92 a došlo tak k rozdělení standardu.

verze RSS	rok vzniku	základní vlastnosti
0.90	1999	RDF model
0.91	1999	bez RDF, jednoduchá specifikace
0.92	2000	bez RDF, stejné jako 0.91
1.0	2000	RDF model, modularita, komplexní a složité
2.0	2002	bez RDF, jmenné prostory XML

Tabulka 2.1: Srovnání jednotlivých verzí RSS

2.1.2.3. RSS 2.0

Na počátku RSS 2.0 byla snaha obou táborů opět sjednotit RSS standard. Bohužel došlo k dalšímu názorovému rozkolu, který se týkal hlavně začlenění datového modelu RDF. Nakonec došlo ke kompromisu a v září 2002 byla vydána specifikace RSS 2.0 jako nástupce verze 0.92 a vývoj RSS byl „zmražen". Hlavní novinkou RSS 2.0 byla možnost používat jmenné prostory XML. Navzdory tomu, že došlo k několika málo úpravám specifikace, nebylo číslo verze již změněno. V roce 2003 byla autorská práva RSS 2.0 předána do rukou Harvardské univerzity, která ji publikovala pod licencí Creative Commons Attribution / Share Alike.

2.2. Historie formátu Atom

2.2.1. Počátky nového formátu

V roce 2003, kdy již téměř nepokračoval aktivní vývoj formátu RSS, se Sam Ruby z firmy IBM pustil do vývoje zcela nového systému pro syndikaci webového obsahu. Navrhl nový syndikační formát a také příslušné API pro vytváření a editování příspěvků na blogu. Pojmenování tohoto formátu vystřídalo několik variant (Pie, Echo, Necho), nakonec ale zvítězilo označení Atom.

V červenci 2003 byla vydána první verze Atom 0.2. Na konci téhož roku bylo zveřejněno nové sestavení Atom 0.3. Tato verze byla první, která se dočkala výrazného rozšíření v nástrojích pro syndikaci obsahu na webu a která se stala součástí několika služeb společnosti Google.

2.2.2. Standardizace Atomu

V roce 2004 vznikla snaha přesunout projekt pod některou z organizací pro správu a vývoj internetových standardů. Nakonec byla vybrána organizace Internet Enginnering Task Force (IETF) a posléze vznikla pracovní skupina Atompub. Výsledkem její práce byl syndikační formát Atom (Atom Syndication Format), který byl v roce 2005 vydán jako standard IETF s označením RFC 4287. O dva roky později pak následoval publikační protokol Atom (Atom Publishing Protocol) vydaný jako standard IEFT označený RFC 5023.

3. Popis formátů pro syndikaci obsahu

V této kapitole se budu věnovat vlastní struktuře formátů pro syndikaci obsahu. Konkrétně se zaměřím na RSS ve verzi 2.0 a 1.0 a Atom 1.0. Záměrně popíši nejdříve formát RSS 2.0, jelikož se jedná o jednodušší a uživatelsky přívětivější z obou verzí RSS. Starší specifikace tohoto formátu (0.91, 0.92 atd.) nebudu rozebírat, protože neexistuje přesná dokumentace a od verze 2.0 se liší jen v maličkostech.

3.1. RSS 2.0

O formát RSS 2.0 [12] se od roku 2003 stará Harvardská univerzita a kompletní specifikace se nachází na adrese http://blogs.law.harvard.edu/tech/rss. Nejedná se ovšem o standard.

3.1.1. Základní struktura

Formát RSS je založen na XML, a proto se jeho struktura řídí pravidly pro zápis XML dokumentů. Popis a vlastnosti formátu XML nejsou součástí této práce a pro další pochopení je vyžadována jejich základní znalost. Nyní již k vlastnímu popisu formátu a jednotlivých elementů RSS 2.0.

Každý XML dokument obsahuje jeden kořenový element, v tomto případě se jedná o element `rss` s atributem `version`. Následuje jediný element `channel`, který uzavírá celý obsah kanálu a přidružená metadata. Konkrétně tedy:

```
<rss version="2.0">
<channel>
  … elementy dle specifikace …
</channel>
</rss>
```

3.1.2. Povinné prvky elementu `<channel>`

Element `channel` obsahuje 3 povinné podelementy:

`title` Název kanálu, většinou shodný s názvem webu.

`link` URL odkazující na přidružený zdroj, většinou webovou stránku. Musí začínat jedním z oficiálně schválených schémat URL.

`description` Stručný popis obsahu kanálu.

Přestože specifikace RSS 2.0 přímo neurčuje, co může být obsahem těchto elementů, je doporučeno používat pouze prostý text bez HTML značek. Oficiálně je HTML obsah povolen pouze v elementu `description`, ale v tomto případě musí být značky kódovány pomocí entit nebo je nutné použít blok `CDATA`, jak je zobrazeno v následující ukázce:

```
<description>
  Toto je &lt;em&gt;levá závorka:&lt;/em&gt; &amp;lt;
</description>

<description>
<![CDATA[Toto je <em>levá závorka</em> &lt;]]>
</description>
```

3.1.3. Nepovinné prvky elementu `<channel>`

Specifikace určuje 16 volitelných elementů, které mohou, ale nemusí být obsahem elementu `channel`. Jedná se o tzv. metadata, která více specifikují daný kanál. Mezi tyto elementy patří:

`language` Specifikuje jazyk, v kterém je kanál napsán, a může obsahovat pouze kódy podle standardu RFC 1766.

`copyright` Autorská práva na obsah zdroje.

`managingEditor` Emailová adresa osoby zodpovědné za obsah kanálu.

`webMaster` Emailová adresa osoby odpovídající za technickou stránku kanálu.

`pubDate` Datum a čas (ve formátu RFC 822), kdy byl obsah publikován.

`lastBuildDate` Datum a čas (ve formátu RFC 822), kdy byl kanál naposledy změněn. Na rozdíl od `pubDate` musí obsah `lastBuildDate` odkazovat do minulosti.

`category` Specifikuje jednu nebo více kategorií, pod které kanál patří.

`generator` Název programu, který vytvořil daný RSS soubor.

`docs` URL odkazující na dokumentaci formátu použitého v daném RSS souboru.

cloud Zřídka používaný element, který umožňuje funkci „Publish and Subscribe[1]“.

ttl `ttl` (Time-to-Live) udává minimální počet minut mezi kontrolami zdroje. Čtečka by neměla zdroj zkontrolovat dříve, než uplyne stanovená doba od poslední kontroly.

image Element popisující doprovodný obrázek náležející ke kanálu. Obsahuje tři povinné a dva nepovinné podelementy:

- `url` Odkaz na daný obrázek.
- `title` Popis obrázku, který odpovídá atributu `alt` uvnitř HTML elementu `img`.
- `link` URL, se kterým je obrázek spojen. Většinou je hodnota stejná jako `channel/link`.
- `width` Nepovinné, udává šířka obrázku v pixelech – max. hodnota 144, výchozí 88.
- `height` Nepovinné, udává výšku obrázku v pixelech – max. hodnota 400, výchozí 31.

rating Ohodnocení obsahu kanálu podle specifikace PICS (http://www.w3.org/PICS/). Používá se jen minimálně.

textInput Element, který umožňuje v RSS parseru zobrazit vstupní pole a tlačítko pro odeslání, ke kterému je asociován CGI skript. Mnoho webů používá tuto funkci například pro vyhledávání v archivu. `textInput` obsahuje čtyři povinné subelementy:

- `title` Popis tlačítka pro odeslání.
- `description` Text vysvětlující funkci elementu `textInput`.
- `name` Jméno objektu, který je předán CGI skriptu.
- `link` Odkaz na CGI skript.

[1] Umožňuje okamžitě zjistit, že byl zdroj aktualizován bez nutnosti opakované kontroly zdroje. Využívá systém notifikace ze strany zdrojového serveru.

skipDays

skipHours Dvojice elementů, které udávají, kdy čtečka nemůže (neměla by) stahovat obsah kanálu. Element `skipDays` může obsahovat až sedm podelementů `day` s následujícím obsahem: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, nebo Sunday. Element `skipHours` může obsahovat až 24 podelementů `hour` reprezentující hodinu v rozmezí 1 – 24. Čtečka by neměla kontrolovat kanál ve dny nebo hodiny uvedené v těchto dvou elementech.

item Obsah kanálu, viz dále.

3.1.4. Element <item>

Soubor ve formátu RSS 2.0 může obsahovat libovolný počet elementů `item`, které tvoří vlastní obsah kanálu. Podle specifikace je tato značka nepovinná, ovšem bez nich nemá RSS zdroj žádný smysl.

Element `item` může obsahovat deset nepovinných podelementů, ovšem za podmínky, že je přítomen alespoň jeden z prvků `item/title` nebo `item/description`. Význam jednotlivých značek je následující:

title Zpravidla se jedná o titulek příspěvku (článku), na který element `item` odkazuje.

link Odkaz na celý příspěvek.

description Obsahuje popis příspěvku. Pravidla pro obsah jsou stejná jako u elementu `channel/description`.

author Emailová adresa autora příspěvku.

category Stejný význam jako element `channel/category`, ovšem vztahuje se pouze k danému `itemu`.

comments Odkaz na stránku, kde může uživatel daný příspěvek okomentovat.

enclosure Popisuje soubor náležící k příspěvku a často se používá pro tzv. podcasting[1]. Nemá žádný vlastní obsah, ovšem obsahuje tři povinné atributy.

- `url` Odkaz na soubor.
- `lenght` Velikost souboru v Bytech.
- `type` Specifikuje MIME typ.

guid Jedná se o zkratku pro anglická slova Globally Unique Identifier. Tento element by měl obsahovat řetězec, který jednoznačně identifikuje daný příspěvek a nesmí se nikdy změnit. Pokud dojde ke změně obsahu elementu `item`, měla by být nastavena nová hodnota `guid`. Tento element může navíc obsahovat nepovinný atribut `isPermalink`, který určuje, zda obsah elementu `guid` může být brán jako odkaz na daný příspěvek. Nahrazuje tak funkci elementu `item/link`. Specifikace ovšem neříká, jak se zachovat, pokud element `item` obsahuje podelementy `link` i `guid`, ale s rozdílným obsahem.

pubDate Datum a čas, kdy byl příspěvek publikován, a platí pro něj stejná pravidla jako pro element `channel/pubDate`.

source Element by měl obsahovat název kanálu, ze kterého položka `item` pochází, a atribut `url` by měl odkazovat na příslušný kanál.

[1] Podcasting je metoda šíření informací do jisté míry konkurující rádiu. Jde o zvukové nebo video záznamy, které autor podcastu umísťuje na Internet v podobě souborů a odkazuje na ně v uzpůsobeném RSS zdroji.

Nakonec ukázka souboru ve formátu RSS 2.0:

```
<?xml version="1.0"?>
<rss version="2.0">
<channel>
  <title>Ukázka RSS 2.0</title>
  <link>http://www.domena.cz/index.php</link>
  <description>Toto je ukázka RSS 2.0</description>
  <language>cs</language>
  <copyright>Copyright 2009, Jan Holub.</copyright>
  <managingEditor>jan.holub@domena.cz</managingEditor>
  <webMaster>webmaster@domena.com</webMaster>
  <pubDate>Fri, 11 Dec 2009 15:00:00 +0100</pubDate>
  <docs>http://blogs.law.harvard.edu/tech/rss</docs>
  <generator>NewsAggregator</generator>
  <ttl>30<ttl>

  <image>
    <title>Příklad RSS 2.0</title>
  <url>http://www.domena.cz/images/logo.gif</url>
    <link>http://www.domena.com/index.html</link>
  </image>

  <item>
    <title>První příspěvek</title>
    <link>http://www.domena.com/news.php?id=1</link>
    <description>Toto je první příspěvek.</description>
    <enclosure url=http://www.domena.cz/001.mp3
  length="543210" type="audio/mpeg"/>
  <comments>http://www.domena.cz/coms.php?id=1</comments>
    <pubDate>Mon, 07 Dec 2009 152:00:00 +0100</pubDate>
    <guid>http://www.domena.com/news.php?id=1</guid>
  </item>

  <item>
    <title>Druhý příspěvek</title>
    <link>http://www.domena.com/news.php?id=2</link>
    <description>Toto je druhý příspěvek.</description>
  <comments>http://www.domena.cz/coms.php?id=2</comments>
    <pubDate>Tue, 08 Dec 2009 152:00:00 +0100</pubDate>
    <guid>http://www.domena.com/news.php?id=2</guid>
  </item>
</channel>
</rss>
```

3.1.5. Rozšiřitelnost pomocí modulů

Moduly lze chápat jako dodatečnou sadu elementů, které rozšiřují možnosti formátu RSS bez toho, aby došlo ke změně základní specifikace. Poskytují tak silný nástroj v případě, že uživatel vyžaduje dodatečné vlastnosti a elementy, které umožní lepší popis jeho syndikovaného obsahu. Nevýhodou ovšem je, že většina agregátorů nebude těmto novým elementům rozumět a bude je ignorovat. Je však možné vytvořit vlastní aplikaci a poté získat určitou výhodu z použitých modulů. Rozšiřitelnost pomocí modulů se týká RSS 2.0 i RSS 1.0.

RSS moduly se vytvářejí pomocí jmenných prostorů XML, které řeší dva základní problémy. Zaprvé umožňují rozlišit význam dvou stejných identifikátorů, které mají ovšem v různém kontextu jiný význam. Je tak možné použít stejné klíčové slovo pro více rozdílných věcí. Tato vlastnost je velmi důležitá, jelikož si kdokoliv může vytvořit vlastní modul a v něm definovat názvy elementů podle svého uvážení, aniž by musel zjišťovat, zda už někdo tato klíčová slova nepoužil v jiném modulu. Druhou vlastností jmenných prostorů je seskupování vzájemně souvisejících prvků. Lze tak snadno vyhledat všechny elementy náležící pod stejný jmenný prostor.

Každý jmenný prostor je určen identifikátorem URI, který by měl být celosvětově jedinečný. Tímto URI je tedy určen význam XML elementů či atributů, a pokud aplikace jmenný prostor nezná, měla by dotyčné značky ignorovat. Jmennému prostoru vždy deklarujeme nějaký prefix, který se uvádí před samotný název značky a je od něj oddělen dvojtečkou.

Mezi nejznámější RSS moduly patří blogChannel Module, Creative Commons Module, Simple Semantic Resolution Module nebo Trackback Module. Následuje jednoduchá ukázka použití fiktivního modulu `music`:

```
<?xml version="1.0"?>
<rss version="2.0" xmlns:music="http://dom.cz/music">
  ...
  <music:interpret>Kabát</music:interpret>
  <music:nazev>Dole v dole</music:nazev>
  <music:rok>2007</music:rok>
  ...
</rss>
```

3.2. RSS 1.0

3.2.1. RDF (Resource Description Framework)

RDF, česky systém popisu zdrojů, tvoří základ pro stavbu kanálů ve formátu RSS 1.0. Jedná se o obecný rámec pro popis, výměnu a znovupoužití dat, který poskytuje jednoduchý model pro zápis metadat a není závislý na konkrétní aplikaci. Zdroje jsou popsány pomocí výroků (statements), přičemž každé tvrzení se skládá z trojice *zdroj-vlastnost-hodnota*. Zdrojem je míněn jakýkoli objekt, který je jednoznačně identifikován pomocí URI a kterému lze přiřadit vlastnost. Zdrojem je tedy například věta v textu identifikovaná pořadovým číslem věty nebo stránka na webu identifikovaná svým URL. Popis zdroje je vyjádřen vlastností, jejíž význam je určen hodnotou vlastnosti. RSS 1.0 je ve své podstatě XML aplikace RDF, kde se dosahuje velké míry modularity pomocí jmenných prostorů XML.

Definujeme-li vlastnost `autor`, pak je možné, že hodnotou této vlastnosti bude například `Jan Holub`. Popis v RDF je založen na definici binární relace (`autor`), která pro zdroj (`text`) přiřadí hodnotu (`Jan Holub`). Relace může přiřadit zdroji jako hodnotu jiný zdroj, který v této relaci hraje roli hodnoty, ale v jiné relaci hraje roli zdroje. Jeden zdroj může mít více relací (vlastností). RDF zápis, který definuje tuto relaci, by mohl vypadat například takto:

```
<rdf:RDF>
  <rdf:Description about="http://dom.cz/zpravodaj/RDF">
    <s:Creator>Jan Holub</s:Creator>
    <s:Title>RDF</s:Title>
    <s:Date>12.12.2009</s:Date>
  </rdf:Description>
</rdf:RDF>
```

V tomto příkladu identifikuje prefix `rdf` ve jménech jednotlivých prvků ty prvky, které jsou deklarovány v rámci specifikace RDF. Tento prefix ovšem není nutné obecně uvádět. Dále je použit jmenný prostor `s`, kde jsou definovány prvky `Creator`, `Title` a `Date`. Každý tento element tvoří vlastnost zdroje a obsah elementu pak hodnotu této vlastnosti. Zdroj je v tomto případě definován pomocí URI `http://dom.cz/zpravodaj/RDF`.

3.2.2. Základy zápisu RDF pomocí XML

3.2.2.1. Kořenový element

Kořenový element RDF dokumentu může obsahovat libovolný počet URI, která identifikují použité jmenné prostory. Vždy ale musí obsahovat tuto deklaraci:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/
                    22-rdf- syntax-ns#">
```

3.2.2.2. Atribut rdf:about

Atribut `rdf:about` definuje URI elementu, pod který spadá. Obsah elementu poté popisuje objekt, který je tímto URI odkazován. Každý zdroj v RDF dokumentu tedy musí mít vlastní atribut `rdf:about`. V následujícím příkladu je zdroj `channel` identifikován pomocí URI `http://www.domena.cz` a má vlastnost `title`, která nabývá hodnoty `Ukázka RDF`.

```
<rdf:RDF xmlns:rdf=http://www.w3.org/1999/02/22-rdf-syntax-
          ns# xmlns="http://purl.org/rss/1.0/">
  <channel rdf:about="http://www.domena.cz">
    <title>Ukázka RDF</title>
  </channel>
</rdf:RDF>
```

3.2.2.3. Atribut rdf:resource

Někdy se stane, že hodnotou některé vlastnosti je jiný zdroj. Aby bylo možné toto popsat, existuje atribut `rdf:resource`. V následující ukázce má zdroj `channel` vlastnost `image`, jejíž hodnotou je další zdroj, v tomto případě `http://www.example.org/obr.jpg`. Pro vlastní popis obrázku je nutné vytvořit element `image` s atributem `rdf:about`.

```
<rdf:RDF xmlns:rdf=http://www.w3.org/1999/02/22-rdf-syntax-
          ns# xmlns="http://purl.org/rss/1.0/">
  <channel rdf:about="http://www.example.org">
    <title>Ukázka RDF</title>
    <image rdf:resource="http://www.domena.cz/obr.jpg" />
  </channel>
  <image rdf:about="http://www.domena.cz/obr.jpg">
    <title>Obrázek něčeho zajímavého</title>
  </image>
</rdf:RDF>
```

3.2.2.4. Kontejnery v RDF

Kontejnery se používají v případě, kdy je potřeba jedné vlastnosti přiřadit více zdrojů. V RDF existují tři odlišné druhy kontejnerů: `rdf:Bag` označuje neuspořádaný seznam, `rdf:Seq` označuje uspořádaný seznam a `rdf:Alt` slouží pro seznam alternativ, kde první položka je výchozí. Každá položka seznamu je pak obsažena v elementu `rdf:li`.

3.2.3. Struktura formátu RSS 1.0

Struktura dokumentu ve formátu RSS 1.0 [9] se od ostatních verzí RSS liší v tom, že elementy `item`, `image` a `textinput` patří do stejné úrovně jako element `channel`, který již neobaluje celý obsah RSS souboru. Tento rozdíl vyžaduje zápis pomocí RDF, aby bylo možné definovat vztahy mezi elementy. Obecná struktura dokumentu vypadá takto:

```
<rdf>
  <channel/>
  <image/>
  <textinput/>
  <item/>
  <item/>
  <item/>
</rdf>
```

3.2.3.1. Povinné prvky elementu `<channel>`

Další úroveň formátu RSS 1.0 začíná vždy povinným elementem `channel`. Ten obsahuje atribut `rdf:about`, který reprezentuje daný kanál. Podle specifikace může být obsahem tohoto atributu URL samotného kanálu nebo URL webu, který reprezentuje.

Element channel může obsahovat mnoho podelementů, ale pouze několik z nich definuje základní specifikace. Zbytek přidávají rozšiřující moduly. Mezi povinné elementy patří:

`title` Titulek kanálu, doporučeno max. 40 znaků.

`description` Popis kanálu, doporučeno max. 400 znaků.

`link` Odkaz na webovou stránku, kterou daný kanál reprezentuje.

Následující elementy jsou povinné jen v případě, že soubor obsahuje objekty, na které tyto elementy odkazují. RSS 1.0 nevyžaduje obrázek, textový vstup ani příspěvky, takže tyto elementy jsou nepovinné.

<image rdf:resource="URI obrázku" />

Tento řádek vytváří vztah mezi elementy `channel` a `image` uvnitř kanálu. URI většinou odpovídá odkazu na daný obrázek.

<textinput rdf:resource="URI textového vstupu" />

Vytváří vztah mezi elementy `channel` a `textinput`.

items

Tento element je velmi důležitý, jelikož definuje vztah mezi elementem `channel` a všemi elementy `item` uvnitř souboru, které tvoří vlastní obsah kanálu. Struktura elementu `items` by měla odpovídat následující struktuře:

```
<items>
  <rdf:Seq>
    <rdf:li resource="URI item 1" />
    <rdf:li resource="URI item 2" />
    ...
  </rdf:Seq>
</items>
```

3.2.3.2. Element <image>

Tento nepovinný, ovšem často užívaný element, odkazuje na obrázek přiřazený ke kanálu. Obsahuje atribut `rdf:resource`, který by měl odkazovat na soubor s obrázkem a tato hodnota by se měla shodovat s hodnotou atributu `rdf:about` elementu `channel/image`.

Element image obsahuje další tři povinné podelementy:

title Text o max. délce 40 znaků, který bude použit jako hodnota atributu `alt` daného obrázku renderovaného jako HTML.

url URL souboru s obrázkem. Hodnota by se měla shodovat s obsahem atributu `rdf:resource`.

link URL, na které bude obrázek odkazovat, pokud bude kanál renderován jako HTML. Zpravidla se jedná o odkaz na stránku, ke které kanál patří.

3.2.3.3. Element `<textinput>`

Tento element obsahuje popis vstupního pole, jehož obsah slouží jako data pro požadavek skrze HTTP. Tohoto se typicky využívá pro vytvoření vyhledávacího pole, pomocí něhož lze například vyhledávat v databázi serveru, ke kterému daný kanál náleží. Element obsahuje atribut `rdf:about`, který by měl mít stejnou hodnotu jako podelement `link`. Mezi povinné subelementy patří:

`title` Text o max. délce 40 znaků pro popis odesílacího tlačítka.

`description` Text vysvětlující funkci vstupního pole. Max. délka 100 znaků.

`name` Název objektu o max. délce 500 znaků, který je předán CGI skriptu.

`link` Odkaz na CGI skript. Max. délka 500 znaků.

3.2.3.4. Element `<item>`

Elementy `item` tvoří vlastní obsah kanálu. Obsahují podrobnosti (popis, metadata atd.) o všech položkách uvedených uvnitř elementu `channel/items`. Na rozdíl od RSS 2.0 mohou prvky `item` uvnitř RSS 1.0 odkazovat na mnoho rozdílných věcí, které mohou být reprezentovány pomocí URL, a to i když se nejedná o klasickou webovou stránku.

Vzhledem k této vlastnosti je mnoho subelementů ovlivněno použitím volitelných modulů. Existují však tři podelementy dané specifikací.

`title` Text o max. délce 100 znaků, který slouží jako název objektu.

`url` URL odkazující na objekt, max. délka 500 znaků.

`description` Nepovinný element, který obsahuje popis objektu. Musí obsahovat prostý text a max. délka je 500 znaků.

Ukázkový soubor ve formátu RSS 1.0 by mohl vypadat například takto:

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://w3.org/1999/02/
                    22-rdf-syntax-ns#">
  <channel rdf:about="http://example.org/index.html">
    <title>Ukázka RSS 1.0</title>
    <link>http://domena.cz/index.html</link>
    <description>Jednoduchý kanál RSS 1.0</description>
    <image rdf:resource="http://domena.cz/obr.jpg" />

    <items>
    <rdf:Seq>
      <rdf:li rdf:resource="http://rss.cz/new.php?id=1" />
      <rdf:li rdf:resource="http://rss.cz/new.php?id=2" />
    </rdf:Seq>
    </items>
  </channel>

 <image rdf:about="http://domena.cz/obr.jpg">
   <title>Zajímavý obrázek</title>
   <link>http://www.domena.cz</link>
   <url> http://domena.cz/obr.jpg</url>
 </image>

  <item rdf:about="http://domena.cz/news.php?id=1">
    <title>První příspěvek</title>
    <link> http://domena.cz/news.php?id=1</link>
  </item>
  <item rdf:about="http://domena.cz/news.php?id=2">
    <title>Druhý příspěvek</title>
    <link>http://domena.cz/news.php?id=2</link>
  </item>
</rdf:RDF>
```

3.2.4. Moduly RSS 1.0

Moduly slouží k rozšíření základní specifikace formátu RSS 1.0, která ve své podstatě obsahuje pouze kostru a základní elementy pro popis syndikovaného obsahu. Další elementy, které jsou například nedílnou součástí standardu RSS 2.0 (metadata týkající se času publikování článků, autorů, správce apod.), je možné přidat právě pomocí modulů [8].

Prakticky od vzniku formátu RSS 1.0 až do současnosti existují tři moduly, schválené jako standard. Jedná se o Dublin Core, Syndication a Content. Dalších více jak 20 modulů se nachází ve fázi návrhu (proposed), ale prakticky se jedná o hotové a bezpečné moduly.

3.2.4.1. Modul Dublin Core (mod_dc)

Tento modul je jedním z nejpoužívanějších rozšíření RSS 1.0. Umožňuje vyjádřit dodatečná metadata, která stanovuje standard Dublin Core. Tento modul je označován prefixem `dc` a odpovídající jmenný prostor je identifikován URI http://purl.org/dc/elements/1.1/.

Všechny elementy modulu Dublin Core jsou volitelné a mohou být použity v elementech `channel`, `item`, `image` a `textinput`. Přestože použití není povinné, je doporučováno vkládat metadata do všech zdrojů. Mezi dostupné elementy patří:

`dc:title`	Název příspěvku (`item`).
`dc:creator`	Jméno autora příspěvku, obvykle s připojeným emailem.
`dc:subject`	Předmět příspěvku.
`dc:description`	Stručný popis příspěvku.
`dc:publisher`	Jméno vydavatele příspěvku (osoba nebo společnost).
`dc:contributor`	Jméno spoluautora příspěvku.
`dc:date`	Datum publikování příspěvku ve formátu W3CDTF **.**
`dc:type`	Povaha příspěvku; musí být jedním z povolených typů, které se nacházejí v seznamu Dublin Core.
`dc:format`	Označení povahy celého kanálu.
`dc:identifier`	Jednoznačný odkaz na zdroj uvnitř daného kontextu. Obdoba atributu `about` elementu `item`.
`dc:source`	Jednoznačný odkaz na zdroj příspěvku. Nemusí se jednat pouze o URL, ale lze použít například ISBN.
`dc:language`	Jazyk, ve kterém je příspěvek napsán.
`dc:relation`	URI souvisejících zdrojů.
`dc:coverage`	Označení oblasti, kterou daný kanál pokrývá. Může se jednat o geografické označení místa, časový úsek nebo označení správní jednotky.
`dc:rights`	Autorská práva vztahující se k obsahu.

3.2.4.2. Modul Syndication (mod_syndication)

Tento velmi používaný modul slouží k informování agregátorů, jak často dochází ke změně zdroje. Lze tak zabránit příliš časté nebo naopak nedostatečné kontrole daného kanálu. Tento modul je označován prefixem `sy` a odpovídající jmenný prostor je identifikován URI http://purl.org/rss/1.0/modules/syndication.

Všechny elementy modulu Syndication se mohou vyskytovat pouze v elementu `channel`. Patří mezi ně:

`sy:updatePeriod` Může obsahovat hodnotu `hourly`, `daily`, `weekly`, `monthly` nebo `yearly`.

`sy:updateFrequency` Číslo udávající kolikrát by měl být zdroj obnoven za danou periodu. Např. pokud je hodnota `updatePeriod` rovna `daily` a `updateFrequency` je 2, potom by měl být kanál kontrolován dvakrát denně.

`sy:updateBase` Datum a čas ve formátu W3CDTF, ze kterého by měly vycházet všechny výpočty.

3.2.4.3. Modul Content (mod_content)

Modul Content umožňuje rozsáhlejší popis syndikovaného obsahu pomocí RDF. Neposkytuje pouze vlastní obsah, ale také velké množství metadat, která lze použít k vytváření rozličných vztahů. Také je nutno poznamenat, že mod_content má jiný význam než element `item/description`, který je často nesprávně používán pro vlastní obsah příspěvku i přes to, že by měl obsahovat pouze zkrácený popis nebo úryvek z obsahu [5].

Modul Content je označován prefixem `content` a odpovídající jmenný prostor je identifikován URI http://purl.org/rss/1.0/modules/content/. Tento modul je komplexnější než ostatní, jelikož má přesně danou strukturu. Skládá se z jednoho hlavního elementu, který obsahuje různé subelementy a atributy, kde některé jsou povinné a jiné ne. Základní struktura vypadá takto:

```
<item>
  <content:items>
  <rdf:Bag>
    <rdf:li>
      <content:item rdf:about="…" >
        <content:format rdf:resource="…" />
        <rdf:value />
      </content:item>
    </rdf:li>
    <rdf:li>
      <content:item>
        <content:format rdf:resource="…" />
        <rdf:value />
      </content:item>
    </rdf:li>
    <rdf:li>
      <content:item>
        …
      </content:item>
    </rdf:li>
  </rdf:Bag>
  </content:items>
</item>
```

Mezi dostupné elementy modulu Content patří:

content:items Obsahuje subelement `rdf:Bag`.

rdf:Bag Obsahuje jeden nebo více subelementů `rdf:li`.

rdf:li Obsahuje povinný subelement `content:item`.

content:item Má vlastní atribut `rdf:about`, který ovšem obsahuje pouze první tento element. Povinné subelementy jsou `content:format` a `rdf:value`. Dále může obsahovat nepovinný element `content:encoding`.

content:format Určuje formát obsahu. Formát je definován pomocí URI, které je obsahem povinného atributu `rdf:resource`.

rdf:value Hodnotou tohoto elementu je vlastní obsah příspěvku. Může se jednat o prostý text nebo XML.

content:encoding Definuje URI reprezentující kódovou sadu.

3.3. Atom 1.0

Atom (Atom Syndication Format) je webový standard pro publikování syndikovaného obsahu, přijatý IETF v prosinci 2005 jako RFC 4287 [10]. Kromě něj byl v říjnu 2007 jako RFC 5023 přijat také Atom Publishing Protocol (zkráceně APP či AtomPub) umožňující vytváření a aktualizaci webových zdrojů ve formátu Atom pomocí HTTP. V této práci se věnuji pouze vlastnímu syndikačnímu formátu.

3.3.1. Základní struktura

Formát Atom obsahuje kořenový element `feed` s atributem `xmlns`, jehož obsahem musí být základní jmenný prostor http://www.w3.org/2005/Atom, který definuje všechny standardizované elementy. Kromě toho zde mohou být definovány i další jmenné prostory XML, což umožňuje použití většiny rozšiřujících modulů pro formáty RSS. Kostra souboru ve formátu Atom 1.0 vypadá takto:

```
<feed xmlns="http://www.w3.org/2005/Atom">
  … metadata …
<entry>
    …
    metadata +
    vlastní obsah
    …
<\entry>
<\feed>
```

Mezi základní vlastnosti formátu Atom patří tyto:

- Všechny časové údaje musejí odpovídat formátu RFC 3339.
- Všechny hodnoty jsou ve formě prostého textu, není-li určeno jinak.
- Jmenný prostor **xml:lang** lze použít pro určení jazyka obsahu.
- Pomocí jmenného prostoru **xml:base** lze určit, jak budou interpretována relativní URI.

3.3.2. Používané konstrukce

Následující konstrukce se týkají jednoho nebo více elementů, které mají společné vlastnosti. Aby nebylo nutné při popisu jednotlivých elementů opakovat stejnou strukturu, budu se odkazovat na jednu z následujících čtyř konstrukcí.

Category

Tato konstrukce se týká jen elementů `category`. Obsahuje jeden povinný a dva nepovinné atributy.

`term` Pojmenování dané kategorie.

`scheme` Kategorizační schéma zadané pomocí URI.

`label` Název kategorie, který je zobrazen uživateli.

Content

Vztahuje se k elementům `content` a `summary`, které obsahují nebo pouze odkazují na vlastní obsah.

Link

Týká se elementu `link`, který má jeden povinný (`href`) a pět volitelných atributů.

`href` URI odkazující na příslušný zdroj (webovou stránku).

`rel` Definuje vztah k souvisejícímu příspěvku. Může nabývat jedné z hodnot `alternate`, `enclosure`, `related`, `self` nebo `via`.

`type` Určuje typ obsahu daného zdroje.

`hreflang` Určuje jazyk daného zdroje.

`title` Textový popisek, obsahující informace o odkazu.

`lenght` Velikost zdroje v Bytech.

Person

Poskytuje popis osoby nebo společnost. Obsahuje povinný element `name` a dva nepovinné elementy `uri` a `email`.

`name` Jméno osoby.

`uri` Domovská stránka osoby.

`email` Emailová adresa osoby.

Text

Tato struktura se týká elementů `title`, `summary`, `content` a `rights`, které zpravidla obsahují krátký text. Atribut `type` definuje, jakým způsobem je informace kódována. Může obsahovat hodnoty `text`, `html`, `xml` a další.

3.3.3. Povinné prvky elementu `<feed>`

Element `feed` obsahuje 4 povinné podelementy:

`id` Identifikátor, který jednoznačně určuje daný zdroj pomocí URI.

`title` Název kanálu.

`updated` Časový údaj, kdy došlo k poslední významné změně zdroje.

`author` Specifikuje autora zdroje a má odpovídající strukturu *Person*. Tento element může existovat vícekrát a je povinný v případě, že ne všechny elementy `entry` obsahují autora.

3.3.4. Volitelné prvky elementu `<feed>`

Specifikace formátu Atom 1.0 definuje celkem devět nepovinných elementů:

`link` Odkazuje na související webovou stránku. Tento element má strukturu *Link* a může se vyskytovat vícenásobně.

`category` Specifikuje kategorii, ke které zdroj patří. Element odpovídá konstrukci *Category* a může se použít vícenásobně.

`contributor` Jména spoluautorů, kteří do zdroje přispívají. Má strukturu *Person* a může se vyskytovat vícenásobně.

`generator` Určuje software, který daný kanál vygeneroval. Může obsahovat dva volitelné atributy `uri` a `version`.

`icon` Odkazuje na malou čtvercovou ikonu, která bude použita pro identifikaci zdroje.

`logo` Odkazuje na větší obrázek, který reprezentuje zdroj.

`rights` Informace o autorských právech.

`subtitle` Textový popis kanálu.

`entry` Element obalující jednotlivé příspěvky.

3.3.5. Element <entry>

Soubor ve formátu Atom 1.0 může obsahovat libovolný počet elementů `entry`, které uvozují vlastní syndikovaný obsah. Jedná se tedy o obdobu elementu `item` u formátu RSS.

Tento element může obsahovat celkem 12 podelementů, z nichž jsou tři povinné a zbytek volitelný. Název i význam mnoha z nich je stejný jako v případě elementu `feed`, rozdíl je pouze v tom, že se vztahují k danému příspěvku a ne k celému zdroji. Jedná se o tyto elementy: `id`, `title`, `updated`, `author`, `link`, `category`, `contributor` a `rights`. Význam zbylých čtyř nepovinných elementů je následující:

`content` Obsah příspěvku nebo odkaz na něj, odpovídá struktuře *Content*.

`summary` Krátké shrnutí, abstrakt nebo výtah z příspěvku. Také se řídí konstrukcí *Content*.

`published` Datum a čas prvního publikování příspěvku.

`source` Pokud je příspěvek (`entry`) kopírován z jednoho zdroje do druhého, měla by být zachována metadata původního zdroje. Konkrétně se jedná o tyto elementy: `contributor`, `author`, `rights` a `category`.

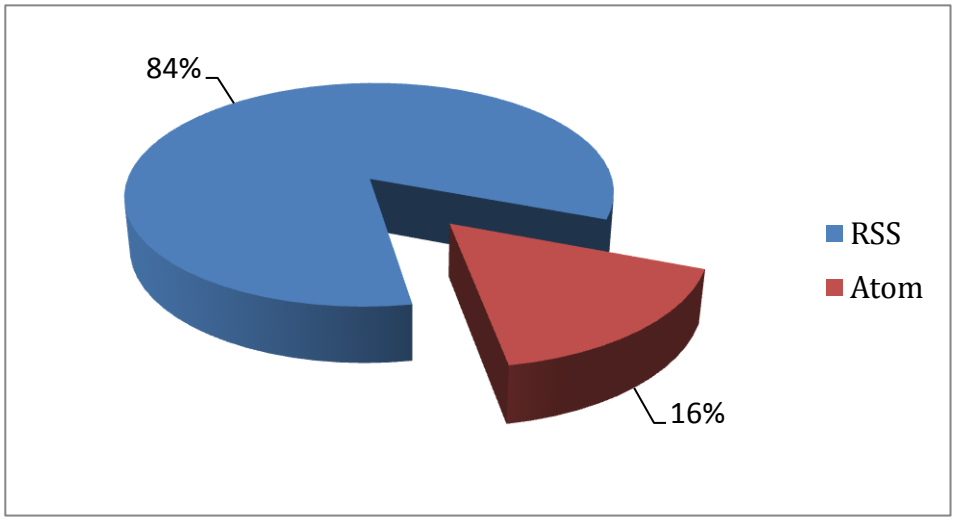
4. Statistika použití syndikačních formátů

V této kapitole se budu zabývat statistikou použití jednotlivých formátů pro syndikaci obsahu na webu. Rozšířenost formátů se pokusím srovnat z hlediska použití v celosvětovém měřítku i se zaměřením pouze na české internetové prostředí, konkrétně na doménu cz.

4.1. Celosvětové statistiky

Při zpracování celosvětových statistik týkajících se četnosti použití jednotlivých syndikačních formátů na webu jsem vycházel z dat dostupných na serverech trends.builtwith.com [4] a syndic8.com [2].

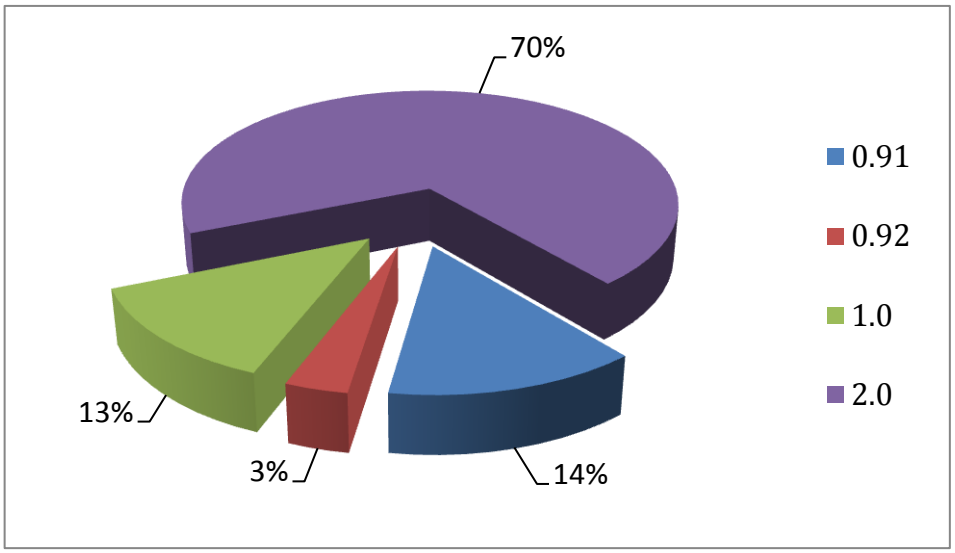
První zmíněná stránka se zabývá získáváním informací a následným vytvářením statistik týkajících se oblíbených webových technologií a jednou z nich je syndikace obsahu. I když jsou detailní statistiky placené, lze všeobecné informace zobrazit přímo na webových stránkách. Provozovatel serveru uvádí, že získaná data vycházejí ze tří milionů indexovaných webů.



Graf 4.1: Statika použití syndikačních formátů na webu

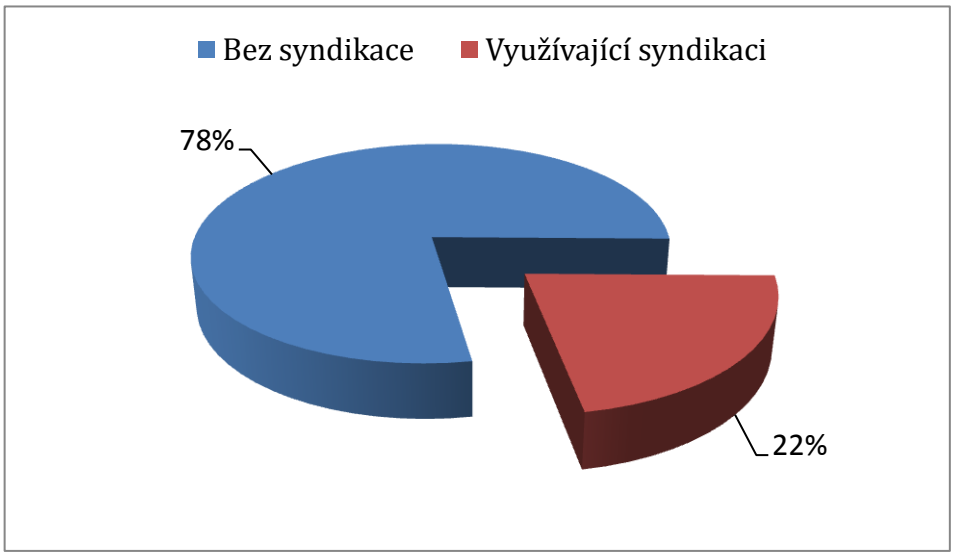
Z těchto statistických údajů vychází Graf 4.1, který zobrazuje rozložení formátů RSS a Atom. Je z něj zřejmé, že RSS zcela vévodí syndikačním formátům používaným na webu. Tuto skutečnost bych připisoval hlavně jeho jednoduchosti v porovnání s Atomem a také tomu, že je starší a více rozšířený v povědomí „běžných“ autorů webových stránek.

Druhý jmenovaný server Syndic8 se zabývá shromažďováním odkazů na různé zdroje syndikovaného obsahu a z nich kromě jiného generuje rozličné statistiky. Stránky jsou bohužel většinu času nedostupné a není tak možné se k informacím dostat. Data je ovšem možné získat pomocí protokolu XML-RPC.



Graf 4.2: Statistika použití jednotlivých verzí RSS

Takto jsem nashromáždil potřebné statistiky, které jsem použil v Grafu 4.2. Z něho je patrné, že nejvíce rozšířenou verzí RSS je poslední verze 2.0 následovaná verzí 0.92. Až na třetím místě je verze 1.0 založená na RDF, která je méně rozšířená nejspíše kvůli své relativní složitosti.

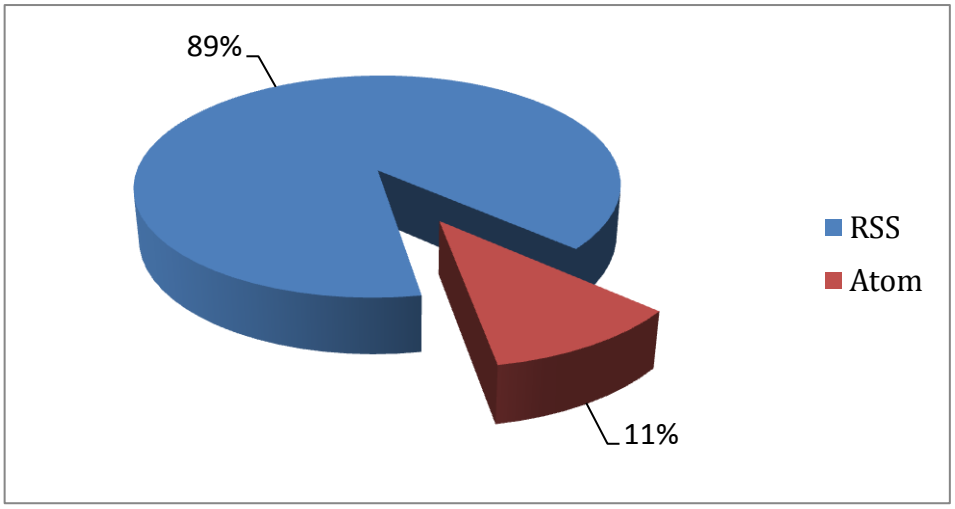


Graf 4.3: Rozšířenost syndikačních technologií na webu

Poslední zobrazený Graf 4.3 ukazuje, jaké procento webů používá některý druh syndikace. Je zřejmé, že syndikaci obsahu využívá asi jen pětina všech webových stránek.

4.2. Domácí statistiky

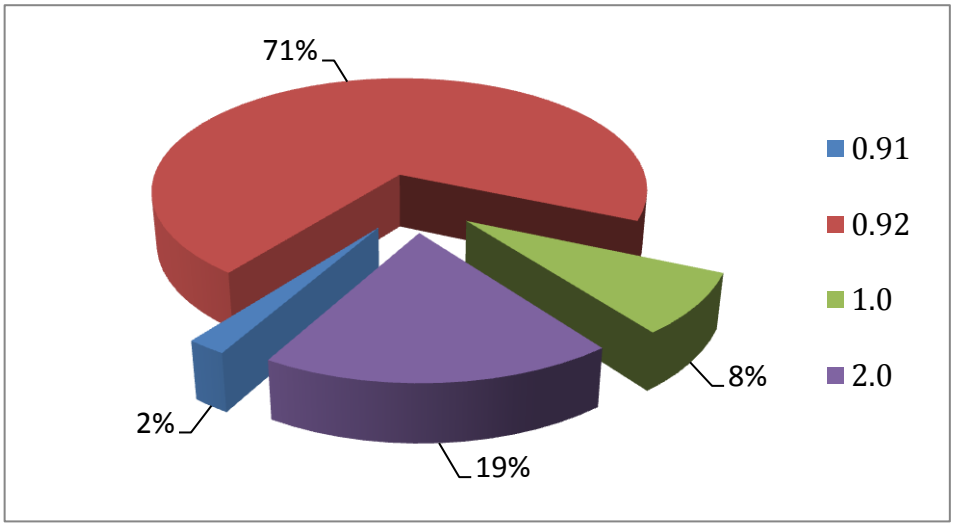
Po prozkoumání globálních statistik jsem se pokusil zmapovat pouze české internetové prostředí (jen doménu cz). Jelikož jsem nikde nenalezl použitelná data, zkusil jsem je získat sám. K tomuto účelu jsem naprogramoval jednoduchý web crawler neboli robota, který automaticky prochází web a snaží se získat odkazy na zdroje syndikovaného obsahu. Každý takto získaný zdroj jsem posléze analyzoval a získal tak potřebné informace pro vytvoření statistik.



Graf 4.4: Rozložení syndikačních formátů na českém webu

Graf 4.4 ukazuje statistiku použití formátů RSS a Atom na českém webu. V porovnání s celosvětovým měřítkem je u nás o něco méně rozšířen formát Atom, což však může vyplývat pouze z malého statistického vzorku.

Velmi odlišné výsledky pak vykazuje statistika použití jednotlivých verzí RSS. V českém prostředí jasně vede verze 0.92 následována verzí 2.0. Tato odlišnost od globálních statistik může být dána opět malým vzorkem zaindexovaných webů. Mohlo dojít například k zahrnutí většího množství zdrojů z blogovacích systémů (Wordpress apod.), kde je použita stejná verze RSS a tím k ovlivnění statistik.



Graf 4.5: Statistika použití jednotlivých verzí RSS na českém webu

5. Aplikace pro správu syndikačních formátů

Tato kapitola se týká praktické části diplomové práce. Budu se v ní zabývat implementací nástroje pro správu syndikačních formátů. Konkrétně se jedná o desktopovou aplikaci napsanou v jazyce Java.

5.1. Vývojové prostředky

5.1.1. Programovací jazyk

Před samotným vývojem aplikace bylo nutné vybrat vhodný programovací jazyk. Jelikož měl být nástroj multiplatformní, zvolil jsem jazyk Java, který rovněž nabízí vhodné prostředky pro vytvoření jednotného grafického rozhraní. Aplikaci je tak možné používat na systémech Windows, Linux, Solaris i Mac OS. Jedinou podmínkou pro spuštění aplikace je přítomnost Java Runtime Enviroment na cílovém počítači.

K vývoji aplikace jsem dále použil vývojové prostředí NetBeans IDE 6.8 a operační systém Windows 7 Professional. Hotový program jsem nakonec testoval na systémech od společnosti Microsoft – Windows XP a Windows 7 – a linuxových distribucích Ubuntu 10.04 a Fedora 12.

5.1.2. Použité knihovny

V aplikaci jsem použil kromě většiny vlastního kódu i několik volně dostupných knihoven. Všechny jsou distribuovány pod některou z licencí pro svobodný software[4]. Knihovny ROME, JDOM a Apache Commons pod Apache licencí a knihovna Substance pod BSD licencí. Není zde tedy žádný problém s případnou další distribucí vytvořené aplikace. Následuje popis jednotlivých knihoven:

- **ROME** – ROME je zkratka pro RSS and Atom Utilities. Jedná se o sadu nástrojů pro práci se syndikačními formáty. Umožňuje především jejich načítání, parsování a generování. Tato knihovna tvoří základ mé aplikace.
- **JDOM** – Knihovnu JDOM vyžaduje pro svoji funkci knihovna ROME. Jedná se komplexní nástroj pro zpracování XML dokumentů.

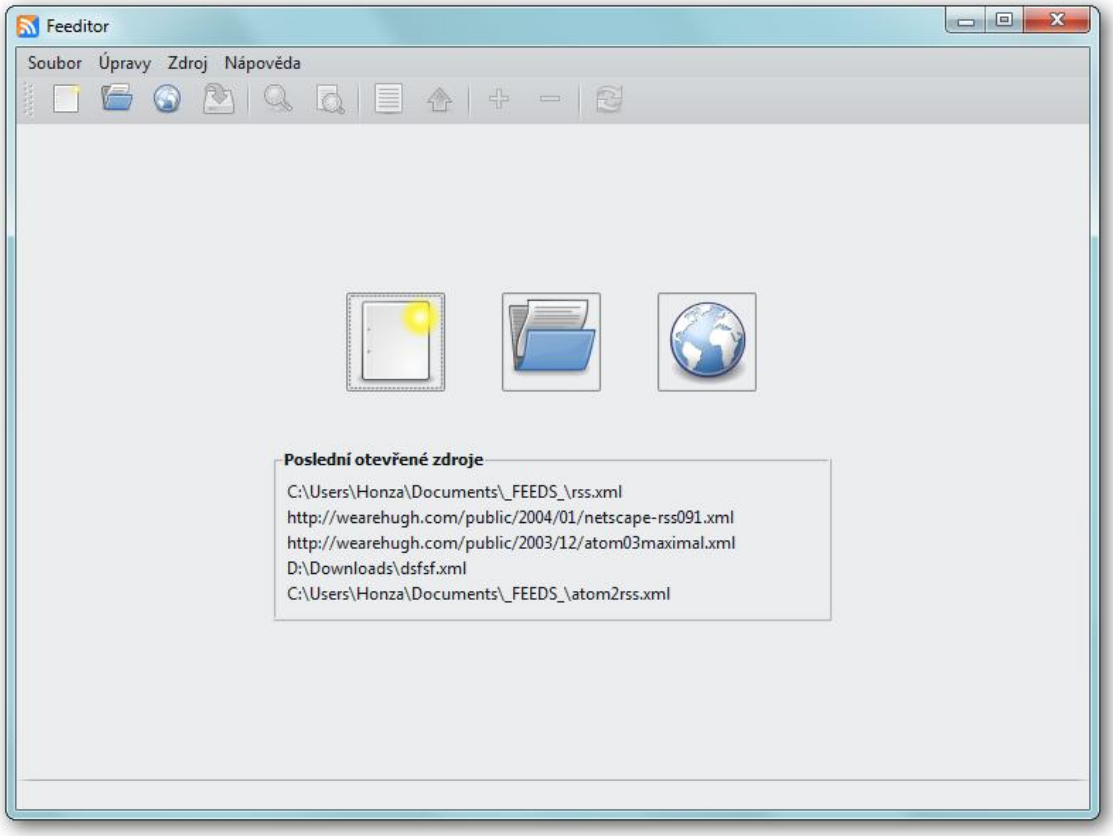
[4] Svobodný software je software, ke kterému je k dispozici také zdrojový kód, spolu s právem tento software používat, modifikovat a distribuovat.

- **Apache Commons Lang** – Knihovna poskytující dodatečné funkce pro třídy z balíčku java.lang.
- **Substance** – Jedná se o knihovnu, která mění vzhled všech grafických komponent obsažených v Javě. Umožňuje, aby aplikace vypadala téměř stejně na všech podporovaných systémech.

5.2. Popis aplikace

Vytvořenou aplikaci jsem pojmenoval **Feeditor**, což je složenina anglických slov *feed* (česky zdroj) a *editor*. Tento název vystihuje podstatu programu, který lze charakterizovat jako jednoduchý editor zdrojů s přidanými funkcemi. Aplikace plně podporuje formáty RSS 0.9x a 2.0 a formáty Atom 0.3 a 1.0. Částečně obsahuje také podporu pro formát RSS 1.0, který lze otevřít a převést do některého z plně podporovaných formátů.

Po spuštění aplikace je k dispozici úvodní obrazovka, která je vyobrazena na Obr. 5.1. Ta nabízí tři základní možnosti: vytvořit nový zdroj, otevřít zdroj z disku a otevřít zdroj z URL. Dále je zde seznam pěti naposledy použitých zdrojů.



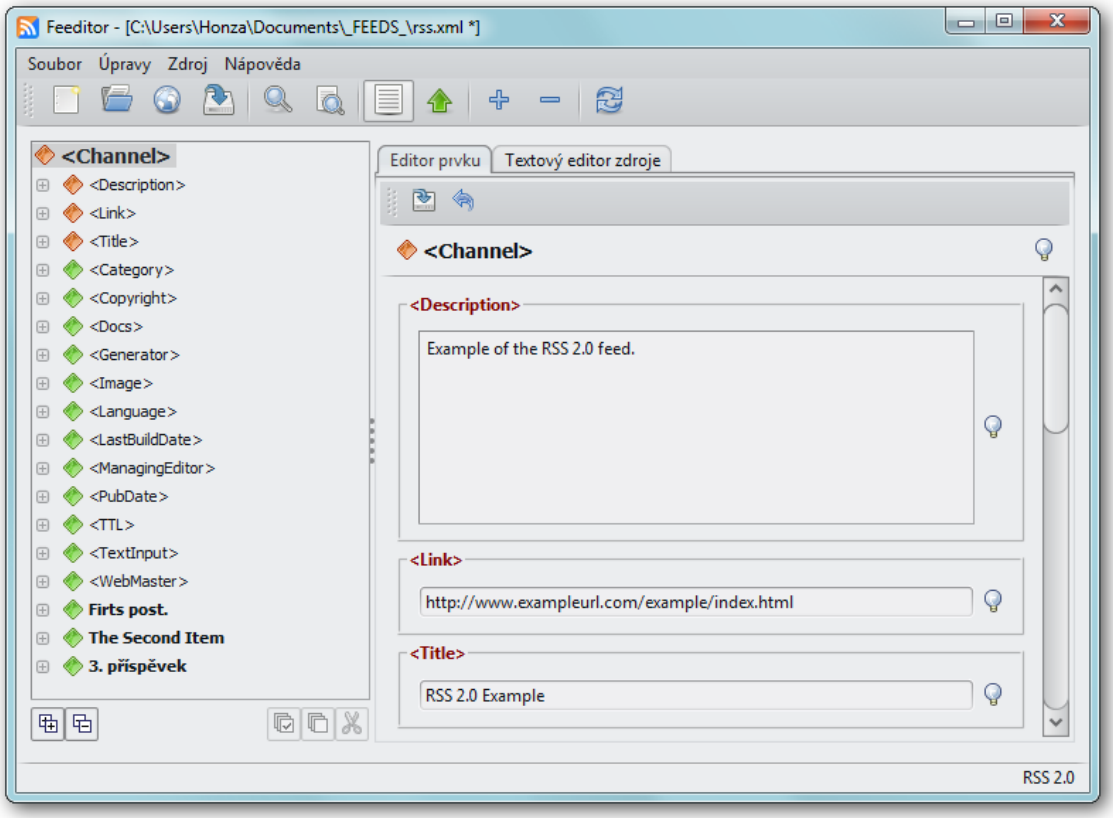
Obr. 5.1: Úvodní obrazovka

5.2.1. Podporované funkce

Feeditor obsahuje následující funkce:

- otevírání zdrojů z disku nebo z URL, seznam naposledy použitých zdrojů
- editace jednotlivých elementů a atributů, jejich přidávání a odebírání
- textový popis každého elementu a atributu
- jednoduchý textový editor zdroje se zvýrazněním syntaxe
- vzájemná konverze podporovaných formátů
- export zdroje do formátu XHTML podle definované šablony
- hledání v otevřeném zdroji

Aplikace s otevřeným zdrojem je zobrazena na Obr. 5.2. Vlevo se nachází stromová struktura, která nejlépe reprezentuje XML dokument, který je základem každého souboru ve formátech RSS či Atom. Po vybrání příslušného elementu je v pravé části okna k dispozici editor obsahu vybraného elementu, případně jeho atributů a podelementů.

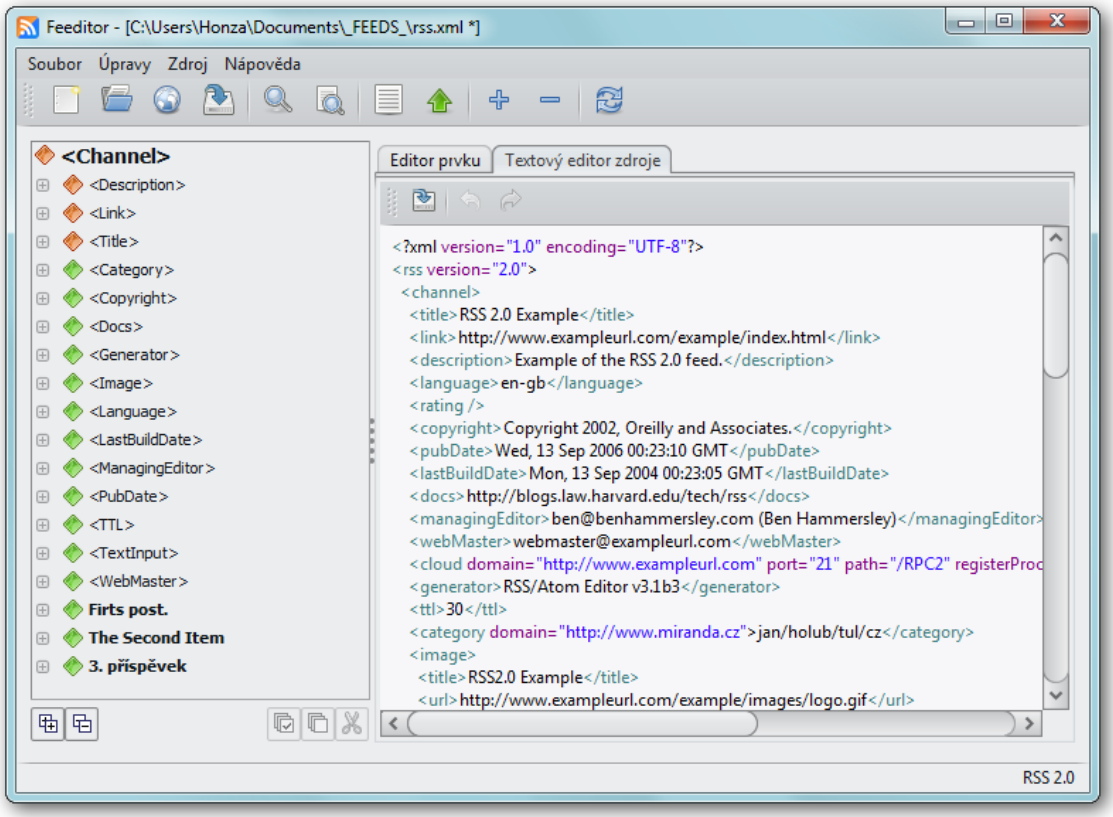


Obr. 5.2: Základní obrazovka s otevřeným zdrojem

Povinné elementy a atributy jsou odlišeny červenou barvou ikony nebo písma. U každého prvku se také nachází malá ikona „žárovky“, nad kterou se po najetí myší, zobrazí textový popis vybraného prvku.

Každé pole pro zadání hodnoty elementu nebo atributu obsahuje omezení odpovídající danému prvku. Pokud například element může obsahovat pouze číselnou hodnotu, nelze do vstupního pole zadat nic jiného než čísla. To samé platí pro prvky, jejichž obsahem může být pouze datum nebo jiná konečná množina hodnot. V tomto režimu editace prvku tedy není možné zadat hodnoty, které by narušily validitu zdroje.

Druhou možností, jak editovat otevřený zdroj, je pomocí jednoduchého vestavěného textového editoru. Ten zatím neobsahuje žádné pokročilé funkce, ale nabízí zvýraznění XML syntaxe a tradiční funkce „Zpět“ a „Vpřed“. Všechny provedené změny se po uložení okamžitě projeví i ve stromu reprezentujícím otevřený zdroj. Navíc v tomto editoru není možné zdroj uložit, pokud není validní.



Obr. 5.2: Textový editor zdroje

5.2.2. Konverze formátů

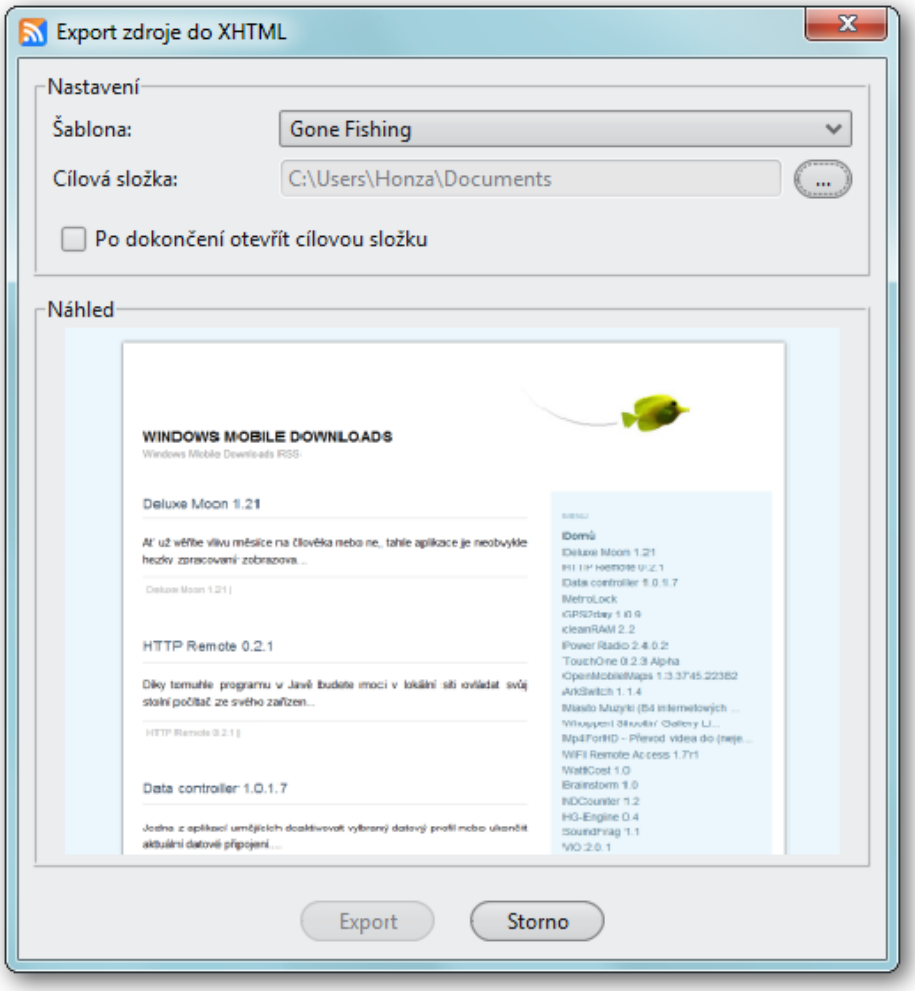
Jednou z hlavních funkcí, které aplikace nabízí, je vzájemná konverze syndikačních formátů. Zdrojem i cílem může být jakýkoliv z podporovaných formátů RSS nebo Atom. Použití je jednoduché. Po nastavení cílového formátu je uživateli zobrazen seznam elementů, které nelze do nového formátu zkonvertovat. Všechny ostatní se převedou podle následující tabulky.

RSS 2.0	Atom 1.0
channel	feed
title	title
link	link
description	subtitle
language	-
copyright	rights
webMaster	-
managingEditor	author
pubDate	-
lastBuildDate	updated
category	category
generator	generator
docs	-
cloud	-
ttl	-
image	logo
-	icon
rating	-
textInput	-
skipHours	-
skipDays	-
item	entry
author	author
-	contributor
description	summary / content
comments	-
enclosure	<link rel="enclosure" …\>
guid	id
source	<link rel="via" …\>

Tab. 5.1: Konverze elementů mezi formáty RSS a Atom

5.2.3. Export zdroje do XHTML

Další z funkcí, které aplikace nabízí, je export zdroje do formátu (X)HTML. Jak vypadá dialogové okno této funkce, je zobrazeno na Obr. 5.3. Je zde možné vybrat cílovou složku a šablonu, podle které budou vytvořeny výsledné html soubory. V dolní části okna je náhled, jak bude zdroj vypadat po otevření v internetovém prohlížeči.



Obr. 5.3: Export zdroje do formátu XHTML

Pro demonstraci této funkce jsem vytvořil dvě šablony, které jsou součástí aplikace. Ty se nacházejí v adresáři „templates“, kde má každá šablona svůj vlastní podadresář. Jelikož nejsou integrovány přímo do programu, může je uživatel snadno upravit dle své vlastní potřeby. Lze také vytvořit zcela nové šablony a to díky jednoduchému formátu, který jsem k tomuto účelu vytvořil.

Šablona je tvořena pěti soubory s příponou html – index.html, entry.html, header.html, footer.html a sidebar.html. Základem každého z nich je klasický (X)HTML kód, který může navíc obsahovat proměnné, jež jsou v průběhu exportu nahrazeny konkrétními hodnotami. Může se jednat o připojení obsahu jiného souboru šablony nebo o skutečná data z exportovaného zdroje. Všechny tyto proměnné jsou uvozeny složenými závorkami.

Základem každé šablony je soubor index.html, který obsahuje kostru úvodní stránky. Ten zpravidla připojuje soubory header.html, footer.html a sidebar.html, které obsahují (X)HTML kód hlavičky, patičky a bočního menu. Podobnou úlohu má soubor entry.html, který však tvoří kostru stránky jednotlivých příspěvků. Definice souboru index.html může vypadat například takto:

```
{header}
<div id="main">
<div id="main-content">
{entries}
  <div class="post">
    <h2><a href="{entry[href]}" title="{entry[title]}">
      {entry[title]}
    </a></h2>
    <div class="post-content">
      {entry[description:100]}
    </div>
    <p>{entry[author]} | {entry[published]}<br /></p>
  </div>
{/entries}
</div>
<div class="sidebar-wrapper">
  {sidebar}
</div>
</div>
{footer}
```

Na tomto příkladu je vidět použití složených závorek pro zápis proměnných. Například proměnná `{header}` zde bude nahrazena obsahem souboru header.html. To samé platí pro `{footer}` a `{sidebar}`. Důležitou dvojicí proměnných je pár `{entries}` a `{/entries}`, které uvozují blok odpovídající jednomu příspěvku. Tento kód se tedy bude ve výsledném souboru opakovat tolikrát, kolik je ve zdroji příspěvků. Jedná se vlastně o jakousi náhradu cyklu `FOR` známého z většiny programovacích jazyků. V bloku `{entries}` se pak mohou vyskytovat proměnné zastupující jednotlivé hodnoty právě exportovaného

příspěvku a to ve formátu `{entry[hodnota]}`. Například `{entry[title]}` bude ve výsledku nahrazena titulkem příspěvku.

Zde je výčet pravidel pro zápis šablon:

- blok `{entries}` je povolen pouze v index.html a sidebar.html
- pro přístup ke globálním hodnotám zdroje se používají proměnné ve tvaru `{feed[hodnota]}`; nahrazovány jsou tyto hodnoty: `title`, `description`, `author`, `copyright`, `generator`, `language`, `link`, `published`, `category`, `href` (odkaz na úvodní stránku)
- pro přístup k hodnotám příspěvku se používají proměnné ve tvaru `{entry[hodnota]}`; nahrazovány jsou tyto hodnoty: `title`, `description`, `author`, `link`, `published`, `updated`, `category`, `href` (odkaz na příspěvek)
- proměnné příspěvku se mohou vyskytovat pouze v bloku `{entries}` nebo v souboru entry.html
- lze použít externí soubor s kaskádovými styly i obrázky
- součástí šablony může být navíc soubor preview.png, který by měl obsahovat její náhled

6. Závěr

Cílem této práce bylo analyzovat metody pro syndikaci obsahu na webu. V současné době existují dvě základní rodiny formátů, které se k tomuto účelů používají. Jedná se o starší formát RSS a novější formát Atom. RSS nebylo původně koncipováno pro syndikaci obsahu, ovšem dnes se jedná o nejrozšířenější metodu, která se k tomuto účelu používá. A to i přes to, že tento formát nebyl nikdy standardizován. Na druhé straně pak stojí formát Atom, jehož aktuální verze 1.0 byla přijata za webový standard jako RFC 4287.

Formát RSS se v současné době dělí na dvě základní verze, které se liší v několika věcech. Jednodušší verze RSS 2.0 poskytuje ucelenou sadu elementů a atributů pro popis syndikovaného obsahu. Bohatě tak postačuje pro distribuci většiny obsahu dostupného na webu. Navíc podporuje rozšiřitelnost pomocí jmenných prostorů XML. Druhou skupinu RSS tvoří verze 1.0, která staví na datovém formátu RDF. Díky tomu poskytuje silnější aparát pro popis rozličného obsahu, ovšem za cenu relativní složitosti zápisu. Vlastní specifikace RSS 1.0 definuje pouze několik základních elementů, další se získávají pomocí modulů.

Formát Atom byl od začátku vyvíjen jako nástroj pro syndikaci obsahu, a proto také poskytuje lepší prostředky pro jeho popis. Atom 1.0 je na rozdíl od RSS 2.0 součástí vlastního jmenného prostoru a lze ho také rozšiřovat pomocí dalších modulů. Mezi hlavní výhody formátu Atom patří explicitní určení typu obsahu uvnitř datových elementů, možnost definovat odlišný jazyk pro různé elementy, schopnost interpretovat relativní URL a lepší sémantika vlastního popisu.

Navzdory tomu, že byl formát Atom navržen přímo pro syndikaci obsahu na webu, je jeho rozšířenost o mnoho menší než u formátu RSS. To bych přičítal tomu, že RSS je starší a zaběhnutý standard, který se mnohem více dostal do podvědomí běžných uživatelů internetu a méně zkušených autorů webových stránek. Podle celosvětových statistik, které jsem zpracoval, je 84 % ze všech syndikačních kanálů postaveno na formátu RSS. Zbylých 16 % pak náleží formátu Atom. Tato čísla vycházející z veřejně dostupných údajů jsou podpořena i vlastnoručně získanými daty. Podle nich je poměr použití obou formátů v prostředí českého internetu ještě vyšší – RSS 89 %, Atom 11 %. Podle další statistiky některý ze způsobů syndikace obsahu využívá zhruba 22 % ze všech webových stránek.

Součástí této práce bylo také vytvořit nástroj pro správu syndikačních formátů. Pojmenoval jsem ho Feeditor a pro vývoj jsem použil jazyk Java, jelikož aplikace měla být multiplatformní. Ve výsledku se jedná o editor zdrojů ve formátech RSS a Atom, do kterého byly implementovány všechny funkce požadované v zadání. Aplikace plně podporuje formáty Atom 0.3 a 1.0 a formáty RSS ve verzi 0.9x a 2.0, částečně pak i verzi 1.0.

Mezi základní funkce aplikace patří otevírání zdrojů z lokálního disku nebo z URL, ukládání zdrojů, přidávání a odebírání elementů, textový editor se zvýrazněním syntaxe nebo vyhledávání v rámci zdroje. Feeditor dále obsahuje podporu pro vzájemnou konverzi podporovaných formátů. Tato funkce automaticky převede odpovídající elementy, případně odstraní prvky, které cílový formát nepodporuje. Další pokročilou funkcí je export zdroje do (X)HTML. Zde je možné definovat vlastní šablony, podle kterých se vytvoří výsledné html soubory. Takto exportovaný zdroj je pak možné otevřít v prohlížeči jako běžnou webovou stránku.

Přestože hotová aplikace obsahuje všechny požadované funkce, je zde ještě prostor pro možná vylepšení. Vhodné by bylo implementovat jednoduchý správce FTP serverů s možností otevírat i ukládat zdroj přímo pomocí FTP. Dále by bylo možné vylepšit nápovědu k jednotlivým elementům a atributům, případně přidat více šablon pro export do (X)HTML. Po vyladění kódu by následovalo vytvoření webové stránku pro prezentaci aplikace, kterou bych následně uvolnil pod některou Open Source licencí.

Seznam použití literatury

[1] AtomEnabled Alliance. *AtomEnabled.org* [online]. 2004, 10/16/2007 [cit. 2010-02-06]. Dostupné z WWW: <www.atomenabled.org>.

[2] BARR, Jeff; KEARNEY, Bill. *Syndic8.com* [online]. 2001 [cit. 2010-02-06]. Dostupné z WWW: <www.syndic8.com>.

[3] BLOCH, Joshua. *Java efektivně: 57 zásad softwarového experta*. Grada Publishing, 2002, ISBN 80-247-0416-1.

[4] *BuiltWith Technology Usage Statistics* [online]. 2010 [cit. 2010-02-06]. Feed Usage Statistics. Dostupné z WWW: <http://trends.builtwith.com/feeds>.

[5] HAMMERSLEY, Ben. *Developing Feeds with RSS and Atom*. O'Reilly Media, 2005, ISBN 978-0-596-00881-9.

[6] HOLZNER, Stephen – ŠINDELÁŘ, Jan. *RSS: Automatické doručování obsahu vašich WWW stránek*. Computer Press, 2007, ISBN 978–80–251–1479–7.

[7] *Java™ Platform, Standard Edition 6 API Specification* [online]. 2006 [cit. 2010-05-10]. Dostupné z WWW: <http://java.sun.com/javase/6/docs/api>.

[8] RSS-DEV Working Group. *Web.resource.org* [online]. 2000-12-09, 2001-03-20 [cit. 2010-05-17]. RDF Site Summary 1.0 Modules. Dostupné z WWW: <http://web.resource.org/rss/1.0/modules/>.

[9] RSS-DEV Working Group. *Web.resource.org* [online]. 2000-12-09, 2008-06-09 [cit. 2010-02-06]. RDF Site Summary (RSS) 1.0. Dostupné z WWW: <http://web.resource.org/rss/1.0/spec>.

[10] *The Atom Syndication Format*. [s.l.] : Internet Engineering Task Force, 2005. 43 s. Dostupné z WWW: <http://www.ietf.org/rfc/rfc4287.txt>.

[11] *TWiki . Javawsxml . Rome* [online]. 2004, Mar/12/2009 [cit. 2010-02-06]. Dostupné z WWW: <http://wiki.java.net/bin/view/Javawsxml/Rome>.

[12] WINER, Dave. *Berkman Center* [online]. July 15, 2003 [cit. 2010-05-17]. RSS 2.0 Specification. Dostupné z WWW: <http://cyber.law.harvard.edu/rss/rss.html>.

[13] ZAKHOUR, Sharon. *Java 6: Výukový kurz*. Computer Press, 2007, ISBN 978-80-251-1575-6.